ChinaXiv合作期刊

基于多尺度残差网络的单应估计方法 *

唐 云[1]，帅鹏飞[1†]，蒋沛凡[1]，邓 飞[1]，杨 强[1, 2]

(1. 成都理工大学 计算机与网络安全学院(牛津布鲁克斯学院)，成都 610059; 2. 成都信息工程大学 控制工程学院，成都 610225)

摘 要：单应估计是许多计算机视觉任务中的一个基础且重要的步骤。传统单应估计方法基于特征点匹配，难以在弱纹理图像中工作。深度学习已经应用于单应估计以提高其鲁棒性，但现有方法均未考虑到由于物体尺度差异导致的多尺度问题，因此精度受限。针对上述问题，提出了一种用于单应估计的多尺度残差网络。该网络能够提取图像的多尺度特征信息，并使用多尺度特征融合模块对特征进行有效融合，此外还通过估计四角点归一化偏移进一步降低了网络优化难度。实验表明，在 MS-COCO 数据集上，该方法平均角点误差仅为 0.788 个像素，达到了亚像素级的精度，并且在99%情况下能够保持较高的精度。由于综合利用了多尺度特征信息且更容易优化，该方法精度显著提高，并具有更强的鲁棒性。

关键词：单应估计；多尺度残差网络；特征融合；四角点归一化偏移；平均角点误差

中图分类号：TP183;TP751 doi: 10.19734/j.issn.1001-3695.2022.03.0124

Homography estimation method based on multi-scale residual network

Tang Yun[1], Shuai Pengfei[1†], Jiang Peifan[1], Deng Fei[1], Yang Qiang[1, 2]

(1. *College of Computer & Network Security*(*Oxford Brookes College*), *Chengdu University of Technology*, *Chengdu* 610059, *China*; 2. *College of Control Engineering*, *Chengdu University of Information Technology*, *Chengdu* 610225, *China*)

Abstract: Homography estimation is a basic and important step in many computer vision tasks. Traditional homography estimation methods are based on feature point matching, which are difficult to work in weak texture images. Deep learning has been applied to homography estimation to improve its robustness, but the existing methods do not consider the multi-scale problem caused by object scale differences, resulting in limited accuracy. To solve the above problems, this paper proposes a multi-scale residual network for homography estimation. The network can extract the multi-scale feature of the image, and used the Multi-Scale Feature Fusion Module to effectively fuse the features. In addition, it further reduced the difficulty of network optimization by estimating the four-corner normalized offset. Experiments on MS-COCO dataset showed that the average corner error of this method was only 0.788 pixels, which achieved sub-pixel accuracy, and can maintain high accuracy in 99% of cases. Due to the comprehensive utilization of multi-scale features and easier to optimize, this method had significantly improved accuracy and stronger robustness.

Key words: homography estimation; multi-scale residual network; feature fusion; four-corner normalized offset; average corner error

chinaXiv:202206.00061v1

0 引言

单应(homography)指从一个平面到另一个平面的可逆映射，这种映射关系可以使用一个 3×3 的非奇异矩阵来表示，其中包含了平移、缩放、旋转与透视，称为单应矩阵[1]。给定两幅图像，从中估计这两幅图像之间的单应变换是计算机视觉中常见的需求。单应估计具有广泛的应用场景，是图像配准[2]、图像拼接[3]、图像矫正[4]、三维重建[5]以及 SLAM[6]等任务中的基础性工作，单应估计的精度对于这些任务有十分重要的影响。

传统的单应估计方法通常是基于特征点匹配的。它使用 SIFT[7]、SURF[8]或 ORB[9]等算法提取图像中的特征点，通过暴力匹配或 FLANN[10]等匹配方法获得两组特征点的对应关系，最后利用 RANSAC[11]算法剔除错误匹配后求解得到单应矩阵。然而这种方法的效果很大程度上依赖于特征点的数量与分布，难以应用于弱纹理图像中，并且步骤比较繁琐，许多超参数都需要人工指定[2]。

随着深度学习的兴起，基于深度学习的单应估计方法被相继提出。2016 年 DeTone 等人[12]首次提出了一种基于 VGG 架构的网络用于单应估计，显示了深度学习方法在单应估计中的潜力；2017 年 Nowruzi 等人[13]使用一种分层堆叠的网络，通过堆叠多个相同网络模块来逐步细化估计结果；Nguyen 等人[14]提出了单应估计的无监督学习方法；2020 年 Zhang 等人[15]以残差网络为主干，并使用内容掩码来选择可靠的区域进行单应估计。这些方法均取得了一定的效果，但都忽略了单应估计的多尺度性。在单应估计中，两次拍摄的照片由于相机的位置、距离和角度的不同，导致两张图像中的同一物体可能具有不同的尺度，而上述网络模型均未考虑到这一点，采用了单一尺度的特征进行单应估计，因此具有一定的局限性。

为了解决单应估计中存在的多尺度问题，同时也受到 SKNet[16]在多尺度特征融合方式上的启发，本文提出了一种

收稿日期：2022-03-13；**修回日期：**2022-05-17 **基金项目：**四川省科学技术厅应用基础项目(2021YJ0086)

作者简介：唐云(1975-)，男，四川成都人，副教授，硕导，硕士，主要研究方向为数值计算、深度学习等；帅鹏飞(1997-)，男(通信作者)，四川眉山人，硕士研究生，主要研究方向为计算机视觉、深度学习等(jerry.tom.cat@qq.com)；蒋沛凡(1997-)，男，江西上饶人，硕士研究生，主要研究方向为计算机视觉、深度学习等；邓飞(1980-)，男，重庆人，教授，硕导，博士，主要研究方向为图像与模式识别、深度学习等；杨强(1988-)，男，四川遂宁人，讲师，硕导，博士，主要研究方向为人工智能、膜计算和特种机器人等.

ChinaXiv合作期刊

多尺度残差单应估计网络(Multi-scale Residual Homography Estimation Network，MRHENet)来进行单应估计。该网络主要创新点有：a)使用不同感受野的卷积层提取多尺度特征进行单应估计；b)提出多尺度特征融合模块(Multi-Scale Feature Fusion Module，MFF Module)来有效融合多尺度特征；c)不直接估计四角点绝对像素偏移[12]，而是估计四角点归一化偏移。在 MS-COCO 数据集[17]与 Apolloscape 数据集[18]上的实验结果表明本文方法优于现有方法。其中，在 MS-COCO 数据集上，本文方法平均角点误差[12]仅为 0.788 个像素，与文献[12]和[15]相比，误差分别降低了 85.0%和 59.4%，因此该方法精度显著提高，并且具有更强的鲁棒性。

1　基本原理

1.1　传统单应估计方法原理

假设通过针孔相机模型对同一平面上的物体进行两次拍摄获得一对图像 A 和 B，那么图像 A 和 B 存在单应变换的关系。使用 3×3 的非奇异单应矩阵 $\boldsymbol{H}$ 来表示这种关系，那么根据单应矩阵的定义[1]，可得单应变换式(1)：

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \boldsymbol{H} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{1}$$

式(1)中单应矩阵 $\boldsymbol{H}$ 将图像 A 上的点(x, y)映射到另一图像 B 上的(x', y')。将式(1)变换后，可得 2 个线性方程：

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}};\ y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}} \tag{2}$$

在单应矩阵 $\boldsymbol{H}$ 中，h_{33} 为非零的缩放系数，一般为 1，因此只有 8 个自由度。根据式(2)，1 组匹配点对可得 2 个线性方程，因此最少只需要 4 组匹配点对即可求解单应矩阵，唯一的限制是这 4 组匹配点对中来自同一图像的点需要满足任意 3 点不共线[1]。

$$\boldsymbol{H} = \begin{cases} \text{无法求解} & ,n < 4 \\ f_{DLT}(\boldsymbol{Corners_A}, \boldsymbol{Corners_B}) & ,n = 4 \\ f_{LS}(\boldsymbol{Corners_A}, \boldsymbol{Corners_B}) & ,n > 4 \end{cases} \tag{3}$$

单应矩阵求解方法如式(3)所示，其中 $\boldsymbol{Corners_A}$、$\boldsymbol{Corners_B}$ 分别表示对两图提取的匹配特征点坐标，n 表示匹配点对的数量。匹配点对若少于 4 组，则无法求解；若只有 4 组，则可以使用直接线性变换法(Direct Linear Transformation，DLT)求解单应矩阵；若多于 4 组，则可以使用最小二乘法(Least Squares，LS)求解。

传统单应估计方法步骤如下：a)通过特征点检测算法从待估计的两图中获取特征点。b)使用特征点匹配算法以建立两组特征点之间的对应关系。c)根据对应关系来求解单应矩阵。针对特征点检测算法，目前已有大量研究：文献[7]提出 SIFT 算法，匹配精度高，但算法复杂度较高，运算时间长；文献[8]对 SIFT 算法运算速度进行了改进，提出 SURF 算法；文献[9]提出了 ORB 算法，计算效率较高但质量不如 SIFT 算法。特征点匹配可使用暴力匹配或 FLANN[10]等方法。由于可能存在误匹配的特征点对，在求解单应矩阵时，还需要使用 RANSAC[11]算法排除误匹配的离群值。

传统单应估计方法依赖于特征点检测质量与分布。实际中，为了达到理想精度而选择的特征点检测算法速度通常较慢，并且对于弱纹理图像，往往难以找到足够多的匹配点对来求解单应矩阵，导致误差很大甚至无法求解。因此，传统单应估计方法鲁棒性较弱，在实际使用时有诸多限制。

1.2　深度学习单应估计方法原理

基于深度学习的单应估计是指通过深度学习方法从输入的两张图像中估计出对应的单应变换，其基本原理如图 1 所示。假设有一对待估计图像 A 和 B，其中 A 为源图像，B 为目标图像，图像 B 是由图像 A 经过单应变换而来，单应矩阵为 $\boldsymbol{H}$。基于深度学习的单应估计方法的基本步骤为：首先对图像 A 和 B 预处理，然后将处理后的图像输入网络，由网络估计出某种形式表示的单应变换，最后计算得到单应矩阵 $\boldsymbol{H}^*$($\boldsymbol{H}^*$表示对 $\boldsymbol{H}$ 的估计值，下文均使用上标“*”表示估计值)。

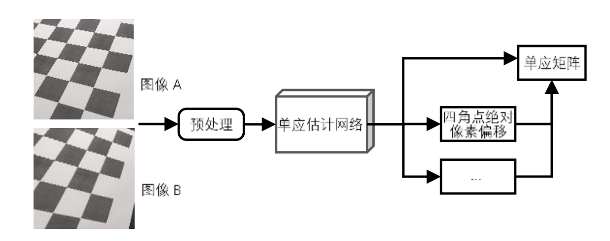


图 1　深度学习单应估计方法原理示意图

Fig. 1　Schematic diagram of homography estimation method based on deep learning

单应变换具有多种表示方式，可以直接采用单应矩阵来表示，也可以采用四角点绝对像素偏移[12]或者其他形式来表示。由于单应矩阵中的各个元素的意义与取值范围各不相同，例如式(1)中 h_{11}、h_{12}、h_{21} 和 h_{22} 表示旋转，h_{13}、h_{23} 表示平移，而平移元素一般会远远大于旋转元素，且又无法对矩阵中的元素进行归一化处理，因此直接使用深度网络估计单应矩阵十分困难。为此，文献[12]不直接估计单应矩阵，而是把单应矩阵参数化为四角点绝对像素偏移，通过网络估计四角点绝对像素偏移从而得到 4 组匹配点对，再使用式(3)中的直接线性法求解以获取单应矩阵。

与传统单应估计方法相比，深度学习单应估计方法在速度和鲁棒性上具有诸多优势。传统方法由于需要检测和匹配特征点，速度通常较慢，并且在弱纹理图像中难以获得稳定有效的匹配点对，导致不能工作。而深度学习方法不需要检测与匹配特征点，因此速度较快。对于传统方法不能处理的弱纹理图像，深度学习同样能根据大量训练数据学习到的规律来估计出较合理的单应矩阵。因此深度学习单应估计方法在实际使用中限制较小，鲁棒性更强，具有较大的应用价值。

2　多尺度残差单应估计网络

2.1　网络结构

2016 年文献[12]首次将一种 VGG 架构的网络用于单应估计，但由于网络结构简单且深度较浅，效果与传统方法相比提升有限。传统的卷积神经网络随着深度不断加深，网络可能会出现退化，训练也会更加困难。因此，2016 年 He 等人[19]提出残差网络(ResNet)，通过恒等映射来降低深度网络训练难度。2020 年文献[15]使用 ResNet34 作为主干，并使用内容掩码来进行单应估计，效果相比于前人有一定提高。但是以上方法均忽略了单应估计中存在的多尺度问题，因此具有一定的局限性。

在单应估计中，两次拍摄的照片由于相机位置、距离和角度的不同，两张图像之间会存在扭曲与缩放，导致图像中的同一物体尺度可能会发生变化，因此单应估计面临多尺度的挑战。为了解决这一问题，本文综合多尺度特征信息来估计四角点归一化偏移，提出了一种多尺度残差单应估计网络来进行单应估计。该网络相比于前人提出的单应估计网络具有明显的创新：首先，网络具有三个多尺度分支，能够提取图像的多尺度特征信息；其次，提出了多尺度特征融合模块(MFF Module)来逐步融合多尺度特征；最后，网络并不直接估计四角点绝对像素偏移，而是估计四角点归一化偏移。网络结构如图 2 所示。

网络输入待估计的两张 128×128 的归一化灰度图像，

chinaXiv:202206.00061v1

ChinaXiv合作期刊

输出表示四角点归一化偏移的 4×2 矩阵 $\boldsymbol{H}^{*}_{4pt_norm}$。具体计算过程如下：首先，将待估计的两图像归一化后堆叠成双通道，同时输入到三个特征提取分支中，分别提取大尺度、中尺度和小尺度的特征。其中，中尺度和小尺度分支具有额外的步长为 2 的卷积层用来减小特征图。三个分支经过 ReLU 激活函数后，大尺度特征图输入到 ResNet34[19]的 stage1 块，stage1 块输出与中尺度特征图通过一个 MFF 模块(缩放系数 r=2)融合后作为 stage2 块的输入，stage2 块输出与小尺度特征图再通过一个 MFF 模块(r=4)融合后依次通过 stage3 块和 stage4 块；最后，特征图通过平均池化后形状变为 1×1×512，再经过全连接层输出 4×2 的矩阵 $\boldsymbol{H}^{*}_{4pt_norm}$。为了加速训练，在每个卷积层后均使用了 BatchNorm 层[20]。

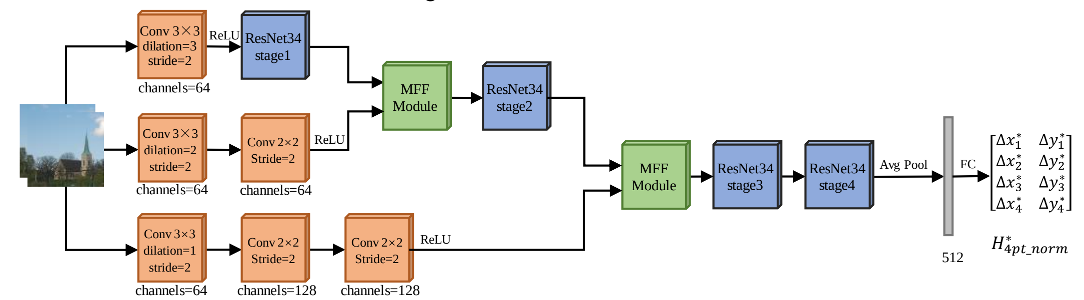


图 2　多尺度残差单应估计网络结构图

Fig. 2　Multi-scale residual homography estimation network structure diagram

2.2　多尺度特征提取

在单应估计中，两次拍摄的照片由于相机位置、距离和角度的不同，两张图像之间会存在扭曲与缩放，导致图像中的同一物体尺度可能会发生变化，因此单应估计面临多尺度的挑战。而文献[12~15]均忽略了这个问题，将两图视为相同尺度对待，使用单一大小的卷积层来提取图像的原始特征。单一的卷积核感受野是固定不变的，导致提取到的特征是在单一空间尺度下的，虽然特征会在后续的卷积层和激活函数后被不断聚合成深层语义特征，感受野逐渐变大，但此时已经丢失了图像原始的空间、几何等细节特征[21]。因此，使用单一尺度的特征来进行单应估计具有一定的局限性，尤其在两张图像具有较大尺度差异时效果不佳。因此，多尺度特征信息对于单应估计是十分重要的。本文把多尺度特征信息引入网络，利用多尺度特征信息来解决单应估计中尺度不一致的问题，从而提高单应估计的精度，使得即使在图像尺度差异较大的情况下该方法也达到理想的效果。

图 2 所示的网络具有大、中、小三个尺度的提取分支，每个分支能够提取对应尺度的特征，因此网络能够利用多尺度特征信息来估计单应变换。具体来说，在三个多尺度分支中，分别使用了感受野为 7×7、5×5 和 3×3 空洞卷积层[22]来提取图像的不同尺度上的特征。图 3 显示了空洞卷积层的原理，与标准卷积相比，空洞卷积可以保证感受野大小不变的同时降低参数量和计算量，能够提高计算效率。

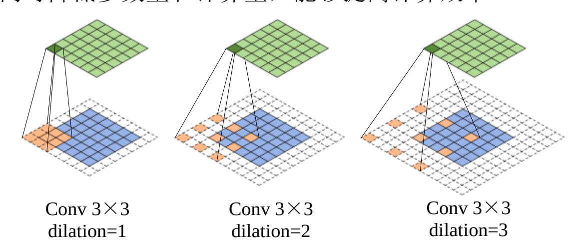


图 3　空洞卷积示意图

Fig. 3　Schematic diagram of dilated convolution

在原始的 ResNet34[19]中，使用了最大池化来对特征图下采样。但是最大池化下采样过程中只保留最大值，导致其余特征信息丢失，因此本文在这里没有使用最大池化，而是将 stage1 块中第一层卷积步长设置为 2(原始步长为 1)，在避免特征信息丢失的同时也减少了计算量。由于后续的 MFF 模块需要输入两个相同形状的特征图，所以在中尺度和小尺度特征提取分支中分别使用了 1 层和 2 层卷积核为 2×2、步长为 2 的卷积层，用于对特征图下采样以匹配后续的 MFF 模块，同时也可以加强特征信息在通道上的交流。

2.3　多尺度特征融合

在基于卷积的单应估计网络中，图像特征通过卷积层逐渐由浅层特征变为深层特征。浅层特征分辨率更高，包含更多位置、几何等细节信息，但是由于经过的卷积层较少，其语义性更低；而深层特征具有更强的语义信息，但是对细节感知能力较差。有效利用浅层特征与深层特征的优势是提高单应估计精度的关键之一。

因此，网络并没有在刚开始就将三种尺度的特征融合，而是在 stage1 块和 stage2 块后分别将中尺度和小尺度的特征融合到网络的主干中。采用了逐步融合的方式，能够利用浅层特征包含的细节信息对深层特征进行补充，实现浅层特征与深层特征优势互补。多尺度分支提取的特征由于尺度不同，如果直接通过相加来融合会导致不同尺度特征混合而难以充分利用多尺度特征的优势；如果将特征在通道上连接，多尺度特征能得到保留，但是通道数就会加倍，计算效率会大幅降低。考虑到特征虽然尺度不同，但均来自于同一输入，所以特征之间会存在冗余。为了充分利用多尺度特征并减少冗余提高计算效率，同时也受到文献[16]在多尺度特征融合方式上的启发，本文提出了使用多尺度特征融合模块 MFF Module 来融合不同尺度的特征。

MFF 模块结构如图 4 所示。输入 2 个不同尺度的特征图 $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^{H\times W\times C}$，MFF 模块输出融合后的特征图 $\boldsymbol{x}_{out} \in \mathbb{R}^{H\times W\times C}$。文献[16]中为了融合不同尺度的特征，先将 $\boldsymbol{x}_1$ 与 $\boldsymbol{x}_2$ 直接相加，再使用 1×1 的平均池化来提取通道上的信息。而本文与文献[16]有两处不同之处：第一，本文先将 $\boldsymbol{x}_1$ 与 $\boldsymbol{x}_2$ 在通道上连接，这样可以保持 $\boldsymbol{x}_1$ 与 $\boldsymbol{x}_2$ 各自的特征，便于后续提取通道上的特征；第二，本文同时使用了 1×1 平均池化与 1×1 最大池化来提取通道上的信息。原因是平均池化只能提取到全局的平均信息不能提取到局部信息，而最大池化只能提取局部信息而不能提取到全局信息，因此同时使用平均池化与最大池化能够综合全局与局部的信息。MFF 模块具体计算过程如下：

a) 将 $\boldsymbol{x}_1$、$\boldsymbol{x}_2$ 在通道上连接，得到 $\boldsymbol{x}_{cat} \in \mathbb{R}^{H\times W\times 2C}$，对 $\boldsymbol{x}_{cat}$ 分别使用 1×1 平均池化和 1×1 最大池化分别提取通道上的信息并将结果相加，得到 $\boldsymbol{x}_g \in \mathbb{R}^{1\times 1\times 2C}$：

$$\boldsymbol{x}_g = AvgPool(\boldsymbol{x}_{cat}) + MaxPool(\boldsymbol{x}_{cat}) \tag{4}$$

b) 使用节点数为 C/r 的全连接层 fc_0(r 表示缩放系数)缩短 $\boldsymbol{x}_g$ 的长度以提高计算效率，随后通过 *ReLU* 函数，得到 $\boldsymbol{z}_r \in \mathbb{R}^{1\times 1\times C/r}$。$\boldsymbol{z}_r$ 分别通过 2 个节点数为 C 的全连接层 fc_1、fc_2，得到 $\boldsymbol{z}_1$、$\boldsymbol{z}_2 \in \mathbb{R}^{1\times 1\times C}$：

$$\boldsymbol{z}_r = \mathrm{ReLU}(fc_0(\boldsymbol{x}_g))$$

chinaXiv:202206.00061v1

ChinaXiv合作期刊

$$z_1 = \mathrm{fc}_1(z_r)\ ;\ z_2 = \mathrm{fc}_2(z_r) \tag{5}$$

c) 将 z_1、z_2 在通道上堆叠，并在通道上使用 *SoftMax* 函数，得到输入的两特征图在通道上的权重 w_1、$w_2 \in \mathbb{R}^{1\times1\times C}$：

$$w_1[i] = \frac{e^{z_1[i]}}{e^{z_1[i]} + e^{z_2[i]}}\ ;\ w_2[i] = \frac{e^{z_2[i]}}{e^{z_1[i]} + e^{z_2[i]}} \tag{6}$$

d) 最后，使用广播乘法将 x_1、x_2 分别与 w_1、w_2 相乘，再将其结果相加，得到融合后的特征图。

$$x_{out} = x_1 * w_1 + x_2 * w_2 \tag{7}$$

不同于将特征图直接简单地相加，MFF 模块能够综合通道上的全局信息为不同尺度的特征图分配相应权重，使得网络具有根据输入的图像选择合适尺度的特征进行单应估计的能力。不同尺度的特征信息经过 MFF 模块融合后能够保留有效特征，减少冗余与无效的特征，有利于网络充分利用多尺度特征信息，从而提高单应估计的精度。

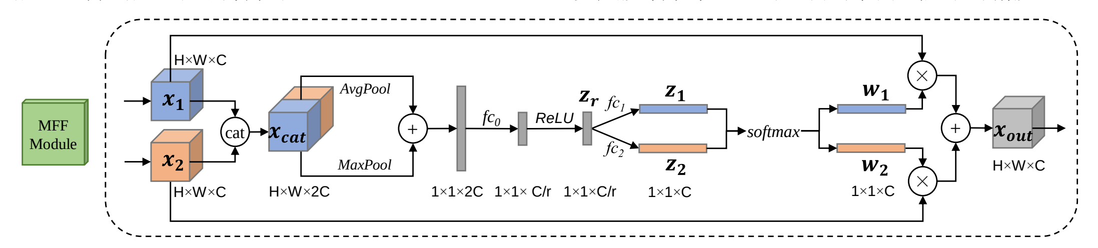


图 4 MFF 模块结构图

Fig. 4 Structure diagram of MFF module

2.4 四角点归一化偏移

文献[12]为了解决直接估计单应矩阵而导致网络难以优化的问题，将单应矩阵参数化为四角点绝对像素偏移 H_{4pt}，通过估计四角点绝对像素偏移来间接估计单应矩阵，在一定程度上降低了网络优化的难度。但是，实际上四角点绝对像素偏移在数值上差异仍然较大，这会使得网络优化过程中梯度差异较大，不利于网络优化。同时也考虑到深度网络中的权重一般会初始化为-1.0~1.0 之间，而四角点绝对像素偏移在大部分情况下会远大于 1 像素，为了学习到这种绝对像素偏移的分布规律，网络权重相对于初始值会发生较大改变，因此使用网络直接估计四角点绝对像素偏移不利于网络收敛。为了进一步降低网络优化难度，本文用网络估计四角点归一化偏移 H_{4pt_norm}，计算方法如式(8)所示。

$$H_{4pt_norm} = \begin{bmatrix} \Delta x_1 & \Delta y_1 \\ \Delta x_2 & \Delta y_2 \\ \Delta x_3 & \Delta y_3 \\ \Delta x_4 & \Delta y_4 \end{bmatrix} = H_{4pt} \cdot \begin{bmatrix} 1/W & 0 \\ 0 & 1/H \end{bmatrix} \tag{8}$$

式(8)中的 Δx_i 与 Δy_i(i=1、2、3、4)表示从图像原点开始顺时针第 i 个点在图像宽度与高度方向上的归一化偏移量，W 与 H 分别表示图像的宽度与高度。由网络估计的四角点归一化偏移 $H^*_{4pt_norm}$ 到单应矩阵 H^* 的计算方法如式(9)~(11)所示。

$$Corners_A = \begin{bmatrix} 0 & 0 \\ 0 & H \\ W & H \\ W & 0 \end{bmatrix} \tag{9}$$

$$Corners^*_B = Corners_A + H^*_{4pt_norm} \cdot \begin{bmatrix} W & 0 \\ 0 & H \end{bmatrix} \tag{10}$$

$$H^* = f_{DLT}(Corners_A, Corners^*_B) \tag{11}$$

3 实验与分析

3.1 网络训练

本文使用 MS-COCO 数据集[17]与 Apolloscape 数据集[18]，按照文献[12]的方法生成实验所需数据集，不同的是本文并没有将图像缩放到 320×240，这会使网络从更少的特征中学习单应估计，有利于增强网络鲁棒性。除此以外，本文还通过将像素值除以 255 的方式来对图像做归一化。总共生成了 22 万对图像，图像尺寸为 128×128，最大角点偏移 ρ=32 像素(图像的四分之一)，其中 18 万对用于训练网络，4 万对用于验证网络。

损失函数使用平均角点误差(Average Corner Error, ACE)[12]，表示预测的四角点偏移与真实值的平均欧式距离，单位为像素(pixel，px)，计算方法如式(12)所示。

$$\mathrm{ACE} = \frac{1}{4}\sum_{i=1}^{4}\sqrt{(\Delta x_i - \Delta x_i^*)^2 + (\Delta y_i - \Delta y_i^*)^2} \times 128 \tag{12}$$

本文基于 Pytorch 深度学习框架来完成实验。训练过程中，使用了概率为 0.5 的随机翻转用于增强数据，采用 Adam 优化器，L2 正则化权重衰减系数设置为 0.003，每次迭代训练 256 对图像，初始学习率为 0.0002，每迭代 20K 次学习率乘以 0.7，总共迭代 200K 次。

3.2 实验测试

为了验证本文方法的实际效果，使用 3.1 节中的方法分别在最大角点偏移 ρ=8px、16px、24px 和 32px 时各生成了 4 万对图像，总共生成了 16 万对图像作为测试集。其中 ρ=8px 表示最大偏移距离较小，ρ=32px 表示最大偏移距离较大，因此测试集中包含了不同程度偏移的图像对。

在测试过程中，平均角点误差 ACE 可能偶尔会出现极端大的情况，导致整个测试集上的平均 ACE(mean average corner error, Mean-ACE)偏高，同时传统方法可能会由于特征点较少而失败。因此本文对 ACE 作出限制，对于 ACE>32px 或者传统方法失败的情况，均视为 ACE=32px。对于 128×128 的图像，如果 ACE>32px 意味结果几乎没有任何价值，所以选择用 32px 作为阈值。由于 Mean-ACE 误差只能反映误差在测试集上的平均情况，不能反映误差分布情况，因此本文引入了中值 ACE(median average corner error, Median-ACE)作为评价指标之一。对于 ACE>32px 或者传统方法失败这两种情况，意味着这次估计是无效的，所以本文还引入了无效率(Invalid Rate，IR)作为评价指标之一，表示无效的情况在测试集中的比例。实验中所有方法均经过多次测试，以避免偶然情况。

为了分别验证本文提出的三个改进点效果，首先进行了消融实验。所有模型均使用相同的方法进行训练与测试，在 MS-COCO 数据集[17]上的消融实验结果如表 1 所示，其中“MFE”表示使用多尺度特征提取，“MFF”表示使用 MFF 模块来融合多尺度特征，“Norm”表示使用了四角点归一化偏移。由表 1 可知，单独使用多尺度特征提取或者四角点归一化偏移均能提升模型效果，并且使用 MFF 模块融合多尺度特征后模型效果有一定提升。当同时使用多尺度特征融合、MFF 模块与四角点归一化偏移时，模型效果能够进一步提升。

在进行了消融实验后，本文使用最终模型与其他方法进行对比实验。参与实验的方法包括了传统方法中的 SIFT[7]+RANSAC[11]法和 ORB[9]+RANSAC[11]法，以及基于深度学习的文献[12, 14, 15]的方法。在 MS-COCO 数据集[17]与 Apolloscape 数据集[18]上的对比实验结果如表 2~3 所示。

由于 Apolloscape 数据集[18]中的图像纹理弱于 MS-

chinaXiv:202206.00061v1

COCO 数据集[17]，因此各种方法在 Apolloscape 数据集上的误差均有一定升高。比较表 2 与表 3，可以明显看出传统方法在较弱纹理图像上误差与无效率大幅增加，这使得传统方法在实际中几乎难以应用。而基于深度学习的方法误差与无效率虽然也有一定升高，但是幅度却较小，这也印证了基于深度学习的方法在弱纹理图像中鲁棒性更强。

表 1　消融实验结果

Tab. 1　Result of ablation experiment

ρ			8px			16px			24px			32px		
MFE	MFF	Norm	Mean-ACE	Median-ACE	IR	Mean-ACE	Median-ACE	IR	Mean-ACE	Median-ACE	IR	Mean-ACE	Median-ACE	IR
			0.654	0.605	**0.00%**	0.758	0.688	**0.00%**	0.978	0.859	**0.00%**	1.431	1.149	0.06%
		√	0.601	0.564	**0.00%**	0.688	0.622	**0.00%**	0.860	0.758	**0.00%**	1.236	0.981	0.05%
√			0.417	0.373	**0.00%**	0.510	0.454	**0.00%**	0.687	0.590	**0.00%**	1.077	0.823	0.02%
√	√		0.352	0.312	**0.00%**	0.430	0.392	**0.00%**	0.570	0.519	**0.00%**	0.881	0.702	0.01%
√	√	√	**0.324**	**0.288**	**0.00%**	**0.395**	**0.352**	**0.00%**	**0.515**	**0.452**	**0.00%**	**0.788**	**0.616**	**0.00%**

表 2　在 MS-COCO 数据集上的对比实验结果

Tab. 2　Results of comparative experiments on MS-COCO dataset

ρ	8px			16px			24px			32px		
评价指标	Mean-ACE	Median-ACE	IR	Mean-ACE	Median-ACE	IR	Mean-ACE	Median-ACE	IR	Mean-ACE	Median-ACE	IR
SIFT[7]+RANSAC[11]	5.339	0.343	13.82%	6.020	0.519	15.18%	6.977	0.764	17.41%	8.179	1.135	20.10%
ORB[9]+RANSAC[11]	12.778	4.960	29.08%	14.262	7.002	31.82%	16.860	11.751	37.88%	20.601	29.483	49.32%
DeTone[12]	2.072	1.779	**0.00%**	2.575	2.189	**0.00%**	3.489	2.885	0.03%	5.252	4.251	0.47%
Nguyen[14]	3.487	2.959	**0.00%**	4.126	3.480	**0.00%**	5.050	4.212	**0.00%**	6.556	5.464	0.12%
Zhang[15]	0.873	0.752	**0.00%**	1.083	0.894	**0.00%**	1.488	1.118	**0.00%**	1.942	1.476	0.05%
Ours	**0.324**	**0.288**	**0.00%**	**0.395**	**0.352**	**0.00%**	**0.515**	**0.452**	**0.00%**	**0.788**	**0.616**	**0.00%**

表 3　在 ApolloScape 数据集上的对比实验结果

Tab. 3　Results of comparative experiments on apolloscape dataset

ρ	8px			16px			24px			32px		
评价指标	Mean-ACE	Median-ACE	IR	Mean-ACE	Median-ACE	IR	Mean-ACE	Median-ACE	IR	Mean-ACE	Median-ACE	IR
SIFT[7]+RANSAC[11]	15.981	7.392	47.68%	16.696	16.493	49.03%	18.001	32.000	52.984%	18.695	32.000	54.73%
ORB[9]+RANSAC[11]	25.559	32.000	74.18%	26.545	32.000	77.05%	27.423	32.000	79.86%	28.845	32.000	85.21%
DeTone[12]	2.265	1.836	**0.00%**	3.050	2.518	**0.00%**	4.114	3.407	0.24%	6.123	4.813	0.62%
Nguyen[14]	3.693	3.010	**0.00%**	4.404	3.769	**0.00%**	5.693	4.693	0.19%	7.596	6.105	0.46%
Zhang[15]	0.967	0.822	**0.00%**	1.177	0.968	**0.00%**	1.623	1.213	**0.00%**	2.315	1.687	0.22%
Ours	**0.348**	**0.298**	**0.00%**	**0.424**	**0.376**	**0.00%**	**0.549**	**0.484**	**0.00%**	**0.922**	**0.664**	**0.01%**

由于基于深度学习的方法在两个数据集上具有相似的趋势，因此本文以 MS-COCO 数据集[17]上的实验结果为例进行分析。图 5 和图 6 分别显示了 MS-COCO 数据集上不同程度偏移下各种方法的 Mean-ACE 误差和 Median-ACE 误差。

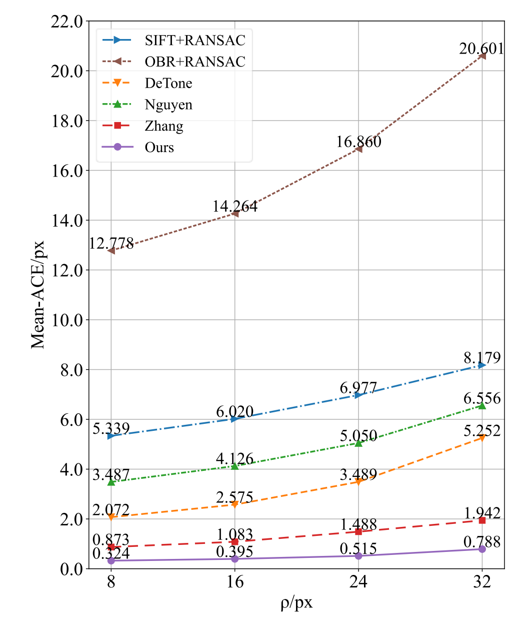


图 5　在 MS-COCO 数据集上不同程度偏移下的 Mean-ACE

Fig. 5　Mean-ACE under different scale offsets on MS-COCO dataset

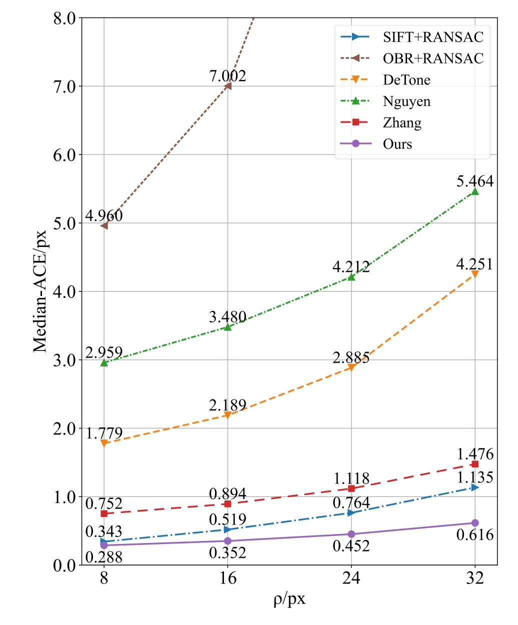


图 6　在 MS-COCO 数据集上不同程度偏移下的 Median-ACE

Fig. 6　Median-ACE under different scale offsets on MS-COCO dataset

从表 2~3 和图 5~6 中可以看出：在传统方法中，SIFT[7]+RANSAC[11]法在精度上明显优于 ORB[9]+RANSAC[11]法。所有方法随着图像最大偏移距离 ρ 由小变大(从 8px 增加到 32px)，Mean-ACE 与 Median-ACE 误差均有不同程度

chinaXiv:202206.00061v1

ChinaXiv合作期刊

地增加。其中，本文误差变化则相对平缓，是唯一能够始终保持亚像素级精度的方法，而其他方法误差增加地比较明显。基于深度学习文献[12, 14, 15]的方法在 MS-COCO 数据集[17]上虽然 Mean-ACE 误差小于 SIFT+RANSAC 法，但是 Median-ACE 误差却比 SIFT+RANSAC 法大，而本文方法则在 Mean-ACE 与 Median-ACE 误差上均领先于 SIFT+RANSAC 法。

图 7 显示了在较大偏移(ρ=32px)时 MS-COCO 数据集[17]上各种方法的 ACE 累积分布曲线。从中可以看出：传统方法中 ORB[9]+RANSAC[11]法表现较差，在大部分情况下都具有相对较高的误差，无效率高达 49.32%；SIFT[7]+RANSAC[11]法表现较好一些，能够在大约 70%的情况下保持较低的误差(ACE<4px)，而在另外 30%的情况下误差会急剧升高，表现变得非常糟糕，无效率为 20.1%。基于深度学习的方法整体上都能在 99%以上的情况下正常工作(ACE<32px)，但文献[12, 14, 15]的方法 60%以上的情况误差高于 SIFT+RANSAC 法，仅能在另外少部分情况下获得比 SIFT+RANSAC 法更好的结果；而本文方法能够在绝大部分情况下具有比 SIFT+RANSAC 法更低的误差，并且能够在 99%情况下保持较高的精度(ACE<4px)，具有最好的鲁棒性。

表 4 显示了不同方法之间的性能对比。在模型大小方面，本文模型比文献[12, 14]更小；在处理速度方面，本文方法速度与传统方法相比具有显著提升，与文献[15]速度相当。

3.3 效果展示

图 8 显示了使用不同方法进行单应估计上的可视化效果。其中最左侧表示被估计的两张图像；右侧图像中的蓝色框与红色框分别表示被估计两图在原图中的位置；绿色框表示使用不同方法估计的结果。红色框与绿色框四角点的平均距离即为 3.1 节中的 ACE 误差，两者越接近则表示误差越低，该方法越好。估计误差显示在对应图像下方，“fail”则表示该方法失败。可以看出 SIFT[7]+RANSAC[11]法与 ORB[9]+RANSAC[11]法在弱纹理图像中几乎不能工作，而本文方法则始终保持较低的误差。

表 4　不同方法的性能对比

Tab. 4　Performance comparison of different methods

方法	模型大小	PPS
SIFT[7]+RANSAC[11]	-	75
ORB[9]+RANSAC[11]	-	100
DeTone[12]	32.61M	10200
Nguyen[14]	31.54M	9800
Zhang[15]	20.31M	5950
Ours	20.46M	5900

*PPS(Pairs Per Second)表示每秒处理的图像对数量。SIFT+RANSAC 与 ORB+RANSAC 运行于 CPU(R5 5600X)，而其他方法运行于 GPU(RTX 3080Ti)。

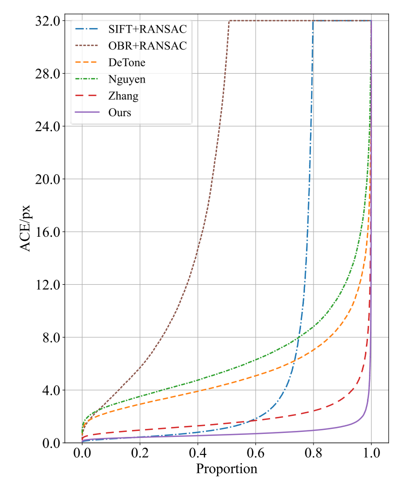


图 7　在 MS-COCO 数据集上的 ACE 累计分布曲线

Fig. 7　ACE cumulative distribution function on MS-COCO dataset

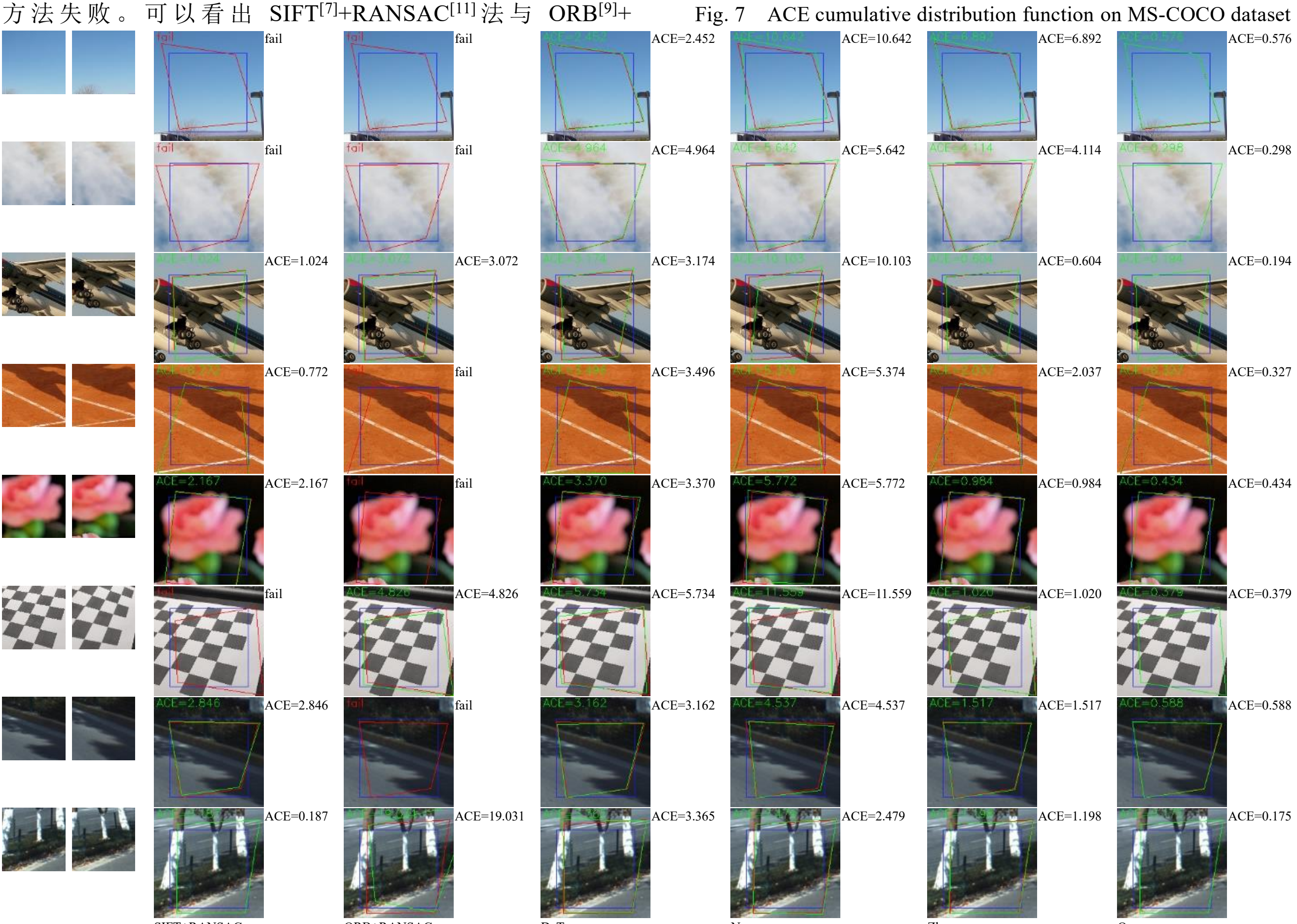


图 8　单应估计效果图

Fig. 8　Effect diagram of homography estimation

chinaXiv:202206.00061v1

ChinaXiv合作期刊

chinaXiv:202206.00061v1

4 结束语

单应估计是图像拼接、图像矫正等许多计算机视觉任务中的一个基础且重要的步骤，具有广泛的应用场景，提高单应估计的精度对这些任务具有重大意义。基于特征点匹配的传统单应估计方法难以在弱纹理图像中工作。然而现有的深度学习方法未考虑到单应估计的多尺度性，使用单一尺度的特征来估计四角点绝对像素偏移，导致图像具有较大偏移时表现不佳。本文提出了一种基于多尺度残差单应估计网络来进行单应估计的方法，通过提取图像的多尺度特征信息并使用 MFF 模块来融合多尺度特性信息，有效利用了多尺度特征信息同时结合了浅层特征与深层特征的优势，并且通过估计四角点归一化偏移来进一步降低了网络优化的难度。在多个数据集上的实验证明了该方法相比于前人提出的传统方法以及深度学习方法精度显著提高，鲁棒性也更强，因此在实际中具有较大的应用价值。

参考文献：

[1] Hartley R, Zisserman A. Multiple view geometryin computer vision [M]. 2nd ed. Cambridge University Press. 2004: 25-48.

[2] 夏丹，周睿. 视差图像配准技术研究综述 [J]. 计算机工程与应用, 2021, 57 (02): 18-27. (Xia Dan, Zhou Rui. Survey of Parallax lmage Registration Technology [J]. Computer Engineering and Applications, 2021, 57 (02): 18-27.)

[3] Brown M, Lowe D G. Recognising panoramas [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2003: 1218.

[4] Yang Xieliu, Yin Chenyu, Tian Dake, *et al.* Rule-based perspective rectification for Chinese text in natural scene images [J]. Multimedia Tools and Applications, 2021, 80 (12): 18243-18262.

[5] Zhang Zhongfei, Hanson A R. 3D reconstruction based on homography mapping [J]. Proc. ARPA96, 1996: 1007-1012.

[6] Davison A J, Reid I D, Molton N D, *et al.* MonoSLAM: Real-time single camera SLAM [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2007, 29 (6): 1052-1067.

[7] Lowe D G. Distinctive image features from scale-invariant keypoints [J]. International Journal of Computer Vision, 2004, 60 (2): 91-110.

[8] Bay H, Tuytelaars T, Van Gool L. SURF: Speeded up robust features [C]// Proc of European Conference on Computer Vision. Berlin: Springer, 2006: 404-417.

[9] Rublee E, Rabaud V, Konolige K, *et al.* ORB: An efficient alternative to SIFT or SURF [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press, 2011: 2564-2571.

[10] Muja M, Lowe D G. Fast approximate nearest neighbors with automatic algorithm configuration [J]. VISAPP (1) , 2009, 2 (331-340): 2.

[11] Fischler M A, Bolles R C. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography [J]. Communications of the ACM, 1981, 24 (6): 381-395.

[12] DeTone D, Malisiewicz T, Rabinovich A. Deep image homography estimation [EB/OL]. (2016) [2022-03-13]. https://arxiv. org/abs/1606. 03798.

[13] Nowruzi F E, Laganiere R, Japkowicz N. Homography estimation from image pairs with hierarchical convolutional networks [C]// Proc of IEEE International Conference on Computer Vision. Piscataway, NJ: IEEE Press. 2017: 913-920.

[14] Nguyen T, Chen S W, Shivakumar S S, *et al.* Unsupervised deep homography: A fast and robust homography estimation model [J]. IEEE Robotics and Automation Letters, 2018, 3 (3): 2346-2353.

[15] Zhang Jirong, Wang Chuan, Liu Shuaicheng, *et al.* Content-aware unsupervised deep homography estimation [C]// Proc of European Conference on Computer Vision. Cham: Springer, 2020: 653-669.

[16] Li Xiang, Wang Wenhai, Hu Xiaolin, *et al.* Selective kernel networks [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2019: 510-519.

[17] Lin T Y, Maire M, Belongie S, *et al.* Microsoft COCO: Common objects in context [C]// Proc of European Conference on Computer Vision. Cham: Springer, 2014: 740-755.

[18] Huang Xinyu, Wang Peng, Cheng Xinjing, *et al.* The ApolloScape Open Dataset for Autonomous Driving and its Application [C]// Proc of IEEE Transactions on Pattern Analysis and Machine Intelligence. Piscataway, NJ: IEEE, 2020: 2702-2719.

[19] He Kaiming, Zhang Xiangyu, Ren Shaoqing, *et al.* Deep residual learning for image recognition [C]// Proc of IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE Press, 2016: 770-778.

[20] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift [C]// Proc of International Conference on Machine Learning. New York: ACM Press, 2015: 448-456.

[21] 姚铭，邓红卫，付文丽，等. 一种改进的 Mask R-CNN 的图像实例分割算法 [J]. 软件, 2021, 42 (09): 78-82. (Yao Ming, Deng Hongwei, Fu Wenli, *et al.* An Improved Mask R-CNN Image Instance Segmentation Algorithm [J]. Computer Engineering & Software, 2021, 42 (09): 78-82.)

[22] Yu F, Koltun V. Multi-scale context aggregation by dilated convolutions [EB/OL]. (2015) [2022-03-13]. https://arxiv. org/abs/1511. 07122.